CLAIM AMENDMENTS

Claims pending:

- At time of the Office Action: Claims 1-37.
- After this Response: Claims 1-4, 6-14, 16-32, and 34-37.

Canceled claims: 5, 15, and 33, without prejudice.

Amended claims: 1, 4, 8, 11, 12, 18, 22, 24, 25, 32, 35, and 37.

New Claims: None.

The listing of claims below will replace prior versions of claims in the application:

1. (Currently Amended) A method comprising:

collecting entropy data, wherein the entropy data includes operating system data;

storing the entropy data in a nonvolatile memory;

updating the entropy data stored in the nonvolatile memory with newly collected entropy data; and

generating a string of random bits from the entropy data stored in the nonvolatile memory.

2. (Original) A method as recited in claim 1 wherein the entropy data is collected from multiple sources.

3. (Original) A method as recited in claim 1 wherein the entropy data is collected from multiple sources within a computer system.

4. (Currently Amended) A method as recited in claim 1 wherein the entropy data includes operating system state information. ~~data related to a processor in a computer system.~~

5. Canceled.

6. (Original) A method as recited in claim 1 wherein the entropy data is maintained in a protected portion of an operating system kernel.

7. (Original) A method as recited in claim 1 wherein the method is executing on a system and the entropy data is inaccessible by an application program executing on the system.

8. (Currently Amended) A method as recited in claim 1 wherein updating the entropy data includes hashing the entropy data stored in the nonvolatile memory with the newly collected entropy data. ~~generating a string of random bits includes hashing the entropy data to generate random seed data.~~

9. (Original) A method as recited in claim 1 wherein updating the entropy data stored in the nonvolatile memory includes collecting new entropy data at periodic intervals.

10. (Original) A method as recited in claim 1 further including communicating the string of random bits to an application program requesting a random number.

11. (Currently Amended) One or more computer-readable memories containing a computer program that is executable by one or more processors, the computer program causing the one or more processors to:

collect entropy data, wherein the entropy data includes processor data;

store the entropy data in a nonvolatile memory;

update the entropy data stored in the nonvolatile memory with newly collected entropy data; and

generate a string of random bits from the entropy data stored in the nonvolatile memory.

12. (Currently Amended) A method comprising:

receiving a request for a random number;

retrieving, from a protected portion of an operating system kernel, ~~nonvolatile memory device~~ entropy data that is regularly updated with newly collected entropy data;

hashing the entropy data to create random seed data;

generating a string of random bits from the random seed data; and

communicating the string of random bits to the requester of the random number.

13. (Original) A method as recited in claim 12 wherein the entropy data is collected from multiple sources within a computer system.

14. (Original) A method as recited in claim 12 wherein the entropy data includes data related to a state of a processor in a computer system and data related to a state of an operating system executing on the computer system.

15. Canceled.

16. (Original) A method as recited in claim 12 wherein the random seed data is maintained in a protected portion of an operating system kernel.

17. (Original) A method as recited in claim 12 wherein the entropy data is inaccessible by the requester of the random number.

18. (Currently Amended) One or more computer-readable memories containing a computer program that is executable by one or more processors, the computer program causing the one or more processors to:

receive a request for a random number;

retrieve entropy data from a protected portion of an operating system kernel ~~nonvolatile memory device~~;

hash the entropy data to create random seed data;

generate a string of random bits from the random seed data; and

communicate the string of random bits to the requester of the random number.

19 (Original) A method comprising:

collecting entropy data;

storing the entropy data in a protected portion of an operating system kernel; and

generating a string of random bits based on the entropy data.

20. (Original) A method as recited in claim 19 wherein the entropy data is collected from multiple sources.

21. (Original) A method as recited in claim 19 wherein the entropy data is inaccessible by an application program.

22. (Currently Amended) A method as recited in claim 19 further comprising updating the entropy data with newly collected entropy data by hashing entropy data in the protected portion of the operating system kernel with the newly collected entropy data.

23. (Original) A method as recited in claim 19 further comprising communicating the string of random bits to an application program requesting a random number.

24. (Currently Amended) One or more computer-readable memories containing a computer program that is executable by one or more processors, the computer program causing the one or more processors to:

~~collecting~~ collect entropy data from multiple sources in a computing system;

~~storing~~ store the entropy data in a protected portion of an operating system kernel; and

~~generating~~ generate a string of random bits based on the entropy data.

25. (Currently Amended) An apparatus comprising:

a nonvolatile memory configured to store entropy data, wherein the entropy data stored in the nonvolatile memory is updated regularly by hashing the entropy data stored in the nonvolatile memory with newly collected entropy data; and

a random number generator, coupled to the nonvolatile memory, to generate strings of random bits using the entropy data received from the nonvolatile memory.

26. (Original) An apparatus as recited in claim 25 wherein the entropy data is collected from multiple sources.

27. (Original) An apparatus as recited in claim 25 wherein the entropy data is updated at periodic intervals.

28. (Original) An apparatus as recited in claim 25 wherein the entropy data is maintained in a protected portion of an operating system kernel such that the entropy data is inaccessible by an application program.

29. (Original) An apparatus as recited in claim 25 wherein the entropy data includes data related to a processor in a computer system and an operating system executing on the computer system.

30. (Original) An apparatus as recited in claim 25 wherein the random number generator hashes the entropy data to generate random seed data.

31. (Original) An apparatus as recited in claim 25 further including a timer coupled to the random number generator, the timer indicating when to update the entropy data stored in the nonvolatile memory device.

32. (Currently Amended) One or more computer-readable media having stored thereon a computer program that, when executed by one or more processors, causes the one or more processors to:

collect entropy data from multiple sources within a computing system, wherein the entropy data is associated with a state of at least one processor;

store the collected entropy data in a nonvolatile memory;

update the entropy data stored in the nonvolatile memory with newly collected entropy data; and

produce a string of random bits from the entropy data stored in the nonvolatile memory.

33. Canceled.

34. (Original) One or more computer-readable media as recited in claim 32 wherein the entropy data is maintained in a protected portion of an operating system kernel.

35. (Currently Amended) One or more computer-readable media as recited in claim 32 wherein the entropy data is associated with ~~includes data related to~~ a state of an operating system executing on the computing ~~a computer~~ system.

36. (Original) One or more computer-readable media as recited in claim 32 wherein to produce a string of random bits from the entropy data, the one or more processors hash the entropy data to generate random seed data.

37. (Currently Amended) One or more computer-readable media as recited in claim 32 wherein the entropy data stored in the nonvolatile memory is updated with newly collected entropy data at periodic intervals by hashing the entropy data stored in the nonvolatile memory with the newly collected entropy data.